数理解析研究所講究録
第 1691 巻 2010 年 85-90

Multiple Random Walk の Cover Time について

穂坂 祐輔 *　　小野 廣隆 †　　山下 雅史 †
* 九州大学大学院システム情報科学府
† 九州大学大学院システム情報科学研究院

1 概要

グラフ上のランダムウォークとは, トークンがグラフ上の頂点をある確率に従ってランダムに遷移していくモデルであり, 要求問い合わせ, 検索, ルーティング, アドホックネットワークや P2P 通信の自己安定化などに応用されている. ランダムウォークの速さの指標のひとつに cover time というものがある. これはトークンがすべての頂点を訪問するまでにかかる遷移数の期待値である. トークンが 1 個の単純ランダムウォーク (simple randpm walk) では, 一般の連結なグラフにおいて cover time の上界が $O(n^2)$ であることが知られている. 本研究では複数のトークンを並列して動かす多重ランダムウォーク (multiple random walk) において $\sqrt{n}$ 個のトークンを用いれば任意の連結グラフで cover time が線形時間となる遷移確率行列が存在することを示した.

2 準備

2.1 単純ランダムウォーク (Simple Random Walk)

グラフ上のランダムウォークとは, トークンがグラフ上の頂点をある確率に従ってランダムに遷移していくモデルであり, 動きまわるトークンが 1 個の単純ランダムウォークを指すことが多い. ランダムウォークの速さを表す指標には hitting time と cover time があり, 以下の用に定義されている.

2.1.1 Hitting Time

グラフ $G=(V,E)$ 上のランダムウォークで, 頂点 $u \in V$ から出発したトークンが遷移確率行列 P に従ってランダムに移動し, 頂点 $v \in V$ に初めて到達するまでに要する遷移数の期待値を $H_G^P(u,v)$ とする. このとき G の Hitting Time(H_G^P) を以下のように定義する.

$$H_G^P = \max_{u,v} H_G^P(u,v)$$

2.1.2 Cover Time

グラフ $G=(V,E)$ 上のランダムウォークで, 頂点 $u \in V$ から出発したトークンが遷移確率行列 P に従ってランダムに移動し, すべての頂点を訪れるに要する遷移数の期待値を $C_G^P(u)$ とする. このとき G の Cover Time(C_G^P) を以下のように定義する.

$$C_G^P = \max_u C_G^P(v)$$

2.1.3 標準ランダムウォーク (Standard Random Walk)

ある頂点を $u \in V$,u に隣接する頂点集合を $N(u)$ とし,u から v への推移確率を $p_{u,v}$ とする. このとき標準ランダムウォークの遷移確率行列 $P = \{p_{u,v}\}_{u,v \in V}$ は以下のように与えられる.

$$p_{u,v} = \begin{cases} \frac{1}{\deg(u)} & (v \in N(u)) \\ 0 & (\text{その他}) \end{cases}$$

2.1.4 単純ランダムウォークの Cover Time の上界

標準ランダムウォークとは各頂点でトークンが隣接頂点に等確率で遷移するランダムウォークである. 標準ランダムウォークの到達時間, 全訪問時間については過去の研究により, 様々なことが解明されている. まず Aleliusu[2] らが, 頂点数 n, 辺数 m の一般のグラフにおいて全訪問時間の上限が $2m(n-1) = O(n^3)$ であることを示している. 次に Aldous[1] が全訪問時間に関する同様の上限を示し, 近年全訪問時間が最大で $4/27n^3$ であることが知られている.[3]

また [3] において, 頂点数 n の木グラフ上の標準ランダムウォークが到達時間, 全訪問時間が共に $O(n^2)$ であることが知られている. 従って任意の連結グラフにおいて, その全域木上で標準ランダムウォークを行うような遷移確率行列を与えることで, 到達時間, 全訪問時間が共に $O(n^2)$ であるようなランダムウォークが設計でき, これは一般の連結なグラフにおける到達時間, 全訪問時間の高速化の上限となっている.

2.2 多重ランダムウォーク (Multiple Random Walk)

多重ランダムウォークとは複数のトークンがそれぞれの遷移確率行列に従い遷移を繰り返すランダムウォークであり,1 ステップですべてのトークンが同時に隣接頂点に遷移する. ここですべてのトークンは同一の頂点を初期位置として, ランダムウォークを始め, いずれかのトークンが頂点を訪問すれば, その頂点は訪問済みとする. また単一のトークンですべての頂点を訪問しなくても良く, 各トークンのランダムウォークは既約なマルコフ連鎖でなくとも良い. (この意味で多重ランダムウォークの遷移確率行列は単純ランダムウォークの遷移確率行列と異なる.)

2.2.1 Cover Time

グラフ $G = (V, E)$ 上の k 個のトークンを使用する多重ランダムウォークで, 頂点 $u \in V$ から出発した k 個のトークンがそれぞれ遷移確率行列 $P_i : 1 \leq i \leq k$ に従ってランダムに移動し, すべての頂点が訪問済みになるまでに要するステップ数の期待値を $C_G^{\{P_i:1\leq i\leq k\}}(u)$ とする. このとき G の Cover Time$C_G^{\{P_i:1\leq i\leq k\}}$ を以下のように定義する.

$$C_G^{\{P_i:1\leq i\leq k\}} = \max_u C_G^{\{P_i:1\leq i\leq k\}}(v)$$

3 高速な多重ランダムウォーク

グラフ上で標準ランダムウォークを行うトークンを使用する多重ランダムウォークについては，最近解析が行われており，一般のグラフにおいて単純標準ランダムウォークに比べ，$\log k \sim k$ 倍の高速化になることが知られている.[3,4,5]
我々はすべてのトークンが標準ランダムウォークに従うのではなく，トークン毎に異なった遷移確率行列を割り当てることで cover time のさらなる高速化ができると考えている．

例えば左右に伸びる頂点数 n の道グラフ P において，すべての頂点で確率 $p_1 \geq \frac{1}{2}$ で右に遷移する遷移確率行列に従うトークン t_1 と，すべての頂点で確率で左 $p_1 \geq \frac{1}{2}$ に遷移する遷移確率行列に従うトークン t_2 が遷移を繰り返す多重ランダムウォークを考える.(図 1)

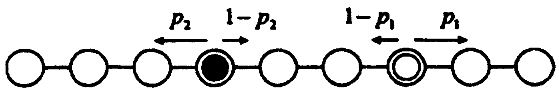

図 1: 道グラフにおける高速な多重ランダムウォーク

この多重ランダムウォーク cover time は t_1 が右の端末点 r,t_2 が左の端末点 l に到達するまでの時間どちらかであり，以下の式で表せる．

$$C_P^{\{P_i:1\leq i\leq 2\}} = \max \max_{u\in V(P)} \{H^{P_1}(u,r)\}, \max_{u\in V(P)} \{H^{P_2}(u,l)\}$$

後述の補題 1 より ,2 つの訪問時間は $O(n)$ であり ,cover time も $O(n)$ である．道グラフ上の単純ランダムウォークの cover Time が $O(n^2)$ なので，トークン 2 個で n 倍の高速化となるこの例は，異なった遷移確率行列に従うトークンを使うことの効能を端的に示している．

補題 1:頂点が $v_1, v_2, ..., v_n$ と並ぶ n 頂点の道グラフ P で遷移確率行列が

$$p_{v_i,v_j} = \begin{cases} p & (j = i+1) \\ 1-p & (j = i-1) \\ 1 & (i = 1, j = 2 ori = n, j = n-1) \\ 0 & \text{otherwise} \end{cases}$$

であるときの頂点 v_i から v_n への hitting time は以下である．

$$H_{v_i,v_n} = \begin{cases} O(n) & (0 < p < \frac{1}{2}) \\ O(n^2) & (p = \frac{1}{2}) \\ O((\frac{1-p}{p})^n) & (\frac{1}{2} < p < 1) \end{cases}$$

証明:道グラフ P 上の頂点 v_i から $v_i + 1$ までの hitting time$H(v_i, v_{i+1})$ は以下になる．

$$\begin{aligned} H(v_i, v_{i+1}) &= p + (1-p)(H(v_{i-1}, v_{i+1}) + 1) \\ &= (1-p)(H(v_{i-1}, v_i) + H(v_i, v_{i+1})) + 1 \end{aligned}$$

これにより以下の漸化式が得られる．

$$H(v_i, v_{i+1}) = \frac{1-p}{p} H(v_{i-1}, v_i) + \frac{1}{p}$$

この漸化式を解くと,

$$H(v_i, v_{i+1}) = 2i - 1$$
$$(p = \frac{1}{2})$$
$$= \frac{1}{2p-1} + (\frac{1-p}{p})^k(\frac{2p-2}{2p-1})$$
$$(p \neq \frac{1}{2})$$

H_{v_i,v_n} はこの i から n までの総和であるので,

$$H(v_i, v_n) = \begin{cases} (n^2 - i^2) & (p = \frac{1}{2}) \\ \frac{n-k}{2p-1} + \frac{(1-p)(1-2p)+p}{p(1-2p)} \cdot \frac{p^2}{(2p-1)(1-p)} \cdot ((\frac{1-p}{p})^{i+1} - (\frac{1-p}{p})^n) & (p \neq \frac{1}{2})) \end{cases}$$

これらの式に p の値を代入して補題の式が得られる.

3.1 訪問する頂点を分担する多重ランダムウォーク

前述の道グラフにおける高速なランダムウォークは僅か2個のトークンで劇的に cover time を短縮できるが, 一般のグラフに対してはこのような高速化は不可能である. ここでは一般のグラフに対する多重ランダムウォークの高速化を述べる.

3.1.1 担当頂点集合 (Assigned Vertex Set)

多重ランダムウォークにおいては, 単一のトークンですべての頂点を訪問する必要はない. そのため各トークンで訪れるべき分担してグラフ上を遷移させることができる. 今, グラフ $G = (V, E)$ 上の多重ランダムのトークンが k 個のトークン $t_i : 1 \leq i \leq k$ を持つとする. このときのトークン t_i が訪れるべき頂点の集合を t_i の担当頂点集合 $V_i \subseteq V$ と定義する. また各 $V_i \subseteq V$ は誘導部分グラフが連結であるとする.

3.1.2 担当頂点集合に対する Cover Time

トークン t_i が担当頂点集合 V_i に含まれる頂点をすべて訪問するのにかかる時間を $C_{V_i}^{P_i}$ と表す. このとき多重ランダムウォーク全体の cover time は以下の式となる.

$$C_G^{\{P_i:1\leq i\leq k\}} = \max_{1\leq i\leq k}\{C_{V_i}^{P_i}\}$$

つまり, すべてのトークンが並列して動いているので, 全体の cover time は担当頂点集合を全訪問するのが最も遅いトークンに依存する. また以下にこの cover time を $O(\max\{n, |V_i|^2\})$ にするような遷移確率行列が存在することを述べる.

3.1.3 部分的な標準ランダムウォーク

トークン $t_i : 1 \leq i \leq k$ の担当頂点集合 V_i が与えられているとする. このとき遷移確率行列 $P_i = p_{u,v}$ を以下を満たすように定める. ただし $Q_i(u)$ は始点を頂点 u とし, 終点を V_i の任意の頂点とする経路, また $\deg_T(u)$ は全域木 T における頂点 u の次数であるとする.

$$P_{u,v} \begin{cases} > \frac{1}{2} & (v \in N(u), u \notin V_i, v \in V(Q_i(u))) \\ < \frac{1}{2} & (v \in N(u), u \notin V_i, v \notin V(Q_i(u)) \\ > \frac{1}{\deg_T(u)} & (v \in N(u), N(u) \not\subseteq V_i, u \in V_i, v \in V_i) \\ < \frac{1}{\deg_T(u)} & (v \in N(u), N(u) \not\subseteq V_i, u \in V_i, v \notin V_i) \\ = \frac{1}{\deg_T(u)} & (v \in N(u), N(u) \subseteq V_i, u \in V_i, v \in V_i) \\ = 0 & (v \notin N(u)) \end{cases}$$

このように遷移確率行列を決めると, 各トークンはそれぞれの割り当て部分グラフに線形時間で遷移する.(補題 1 の拡張) また各トークンは割り当てられた部分グラフ内で標準ランダムウォークとなる.

4 線形 Cover Time を持つ多重ランダムウォーク

定理: 任意の連結グラフにおいて,cover time が線形となる, トークン数 $\sqrt{n}$ の多重ランダムウォークが存在する.(後述の補題 2 を使う)

証明 : 連結グラフ G の全域木 T に対し, 補題 2 を適応し, 頂点数 $O(\sqrt{n})$ の T の誘導部分グラフを $\sqrt{n}$ 個作る. この誘導部分グラフがそれぞれのトークンの担当頂点集合の全域木となるように部分的な標準ランダムウォークを行うように遷移確率行列を定める. 部分的な標準ランダムウォークの性質より, それぞれのトークンは, 初期位置によらず線形時間で各々の担当頂点集合へと移動する. また担当頂点集合の全域木上でトークンは標準ランダムウォークとなり, 担当以外の場所に遷移しても, 定数の期待時間で再び割り当て部分グラフに戻る (補題 1) ので, 各トークンが担当頂点を全訪問するまでの時間は担当頂点集合の全域木上の標準ランダムウォークの cover time で近似できる. このとき多重ランダムウォークの cover time は $O(n + \max_{1 \leq i \leq k} C_{std}(S_i)) = O(n + (\sqrt{n})^2) = O(n)$ となる.

補題 2 : 頂点数 n の木 T の頂点集合は $\sqrt{n}$ 個の頂点数 $O(\sqrt{n})$ の連結部分グラフの頂点集合の和集合で表現できる.

証明 : 木 T で根から開始する深さ優先探索を考え, そのなかで通った頂点の系列を抜き出す.(図 2) すべての頂点は次数に等しい回数出現するので, 系列の長さは $2n - 1$ である. この系列を頂点数 $2\sqrt{n}$ 毎に区切り, いくつかの頂点集合を作る. ここで出来る頂点集合の数は $\sqrt{n}$ 個であり, その誘導部分グラフは走査の性質から連結であり, 頂点数は $O(\sqrt{n})$ である.

5 まとめ

本稿では頂点数 n の任意の連結なグラフにおいて, トークン数 $O(\sqrt{n})$ で cover time が n の線形時間となる多重ランダムウォークが存在することを示した. これは線形な cover time を持つ多重

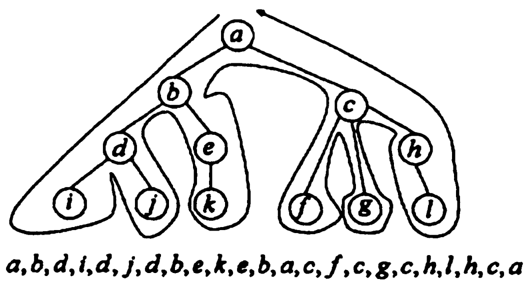

図 2: 深さ優先探索で得られる頂点系列

ランダムウォークのトークン数の上界なっている．下界についても上界と一致して $\Omega(n)$ であると予想している．しかしながら，トークンに訪問する頂点を分担させる以外に，より優れた全訪問のための戦略があるとすれば，上下界ともにさらに下げれる余地がある．

参考文献

[1] D.J.ALdous: On the time taken by random walks on finate groups to visit every state. Z.Wahrsoh.verw.Gebiete62 361(1983)

[2] R.Aleliunas, R.M Karp, R.J Lipton, L.Lovaasz and C.Rackoff: Random walks,universal traversal sequences, and the complexity of maze problems. Proc.20th IEEE Ann.Symposium on Foundations of Computer Science, 218-223(1979)

[3] Noga Alon, Chen Avin, Michal Koucký, Gady Kozma, Zvi Lotker, Mark R.Tuttle: Many Random Walks Are Faster Than One. ArXiv e-prints 705(2007)

[4] Klim Efremenko, Omer Raingold: How Well Do Random Walks Parallelize. APROX and RANDOM 2009,pp.476-489(2009)

[5] Robert Elsässer, Thomas Sauerwald: Tight Bounds for the Cover Time of Multiple Random Walks. ICALP 2009,p.415-426(2009)

[6] Satoshi Ikeda, Izumi Kubo, Norihiro Okumoto, and Masafumi Yamashita: Impact of Local Topological Information on Random Walks on Finate Graphs. Proceedings of Thirtieth International Colloquium on Automata, Languages and Programming, pp.1054-1067(2003)